Journal for Geometry and Graphics
Volume 1 (1997), No. 1, 41–49

Geometric Problems in Computerized Preoperational Planning of a Robot Assisted Total Knee Replacement

O. Tebeleva, G. Elber, Y. Charit, M. Roffman*

Technion, Israel Institute of Technology
Haifa 32000, Israel

* *Also "Carmel" Medical Center, Haifa*

Abstract. A computer graphics system is developed enabling the surgeon to pre-plan a total knee replacement. A series of geometric problems had to be solved in order to introduce a device independent set of coordinates serving the pre-planning as well as the robot that will assist the surgeon. A special efficient algorithm is used for interactive planning and the improvement of the performance of bone cuts.

1. Introduction

Total knee replacement is a widely used orthopedic operation. With a view to the longest possible lifespan for the treated joint it is important to achieve higher implantation accuracy, taking into consideration the anatomy and biomechanism of the joint. Of great help to the surgeon in this direction will be an improved three dimensional computerized preoperative planning system supported by a robot with high accuracy performance. Developing a computer graphic system for this purpose we had to solve a series of geometric problems and to create efficient algorithms. Some of them are set forth in this paper.

In recent years, volumetric data was widely exploited by medical imaging based applications. Medical scanner, such as CT, allows one to sample the Euclidean space, obtaining a three dimensional data set, $\mathcal{D}$. Numerous techniques were developed to extract surfaces of constant iso-value out of $\mathcal{D}$, also known as iso-surfaces. The most widespread method, the Marching Cubes (MC) algorithm, was originally presented in [8]. The MC algorithm processes voxels in $\mathcal{D}$ and isolates the ones that deviate from the desired iso-surface level. Then, each isolated voxel is further processed to extract a polygonal approximation of its intersection with the iso-surface, using a table driven mechanism. The complete set of polygons that results, approximates the desired iso-surface.

In the presented work, the MC algorithm was utilized to obtain a model of the femur out of a CT scan. Thus, the femur model we are dealing with further is represented by a polygonal mesh (Figure 1).

ISSN 1433-8157/$ 2.50

Figure 1: Three views of the femur reconstructed from CT data.

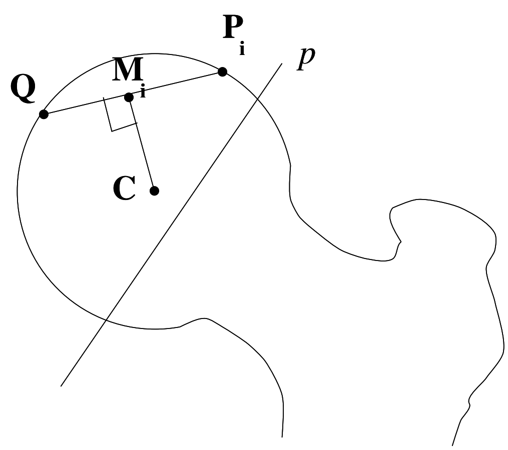


Figure 2: Searching for the femoral head's center

2. Determination of Object Orientation

Assume the leg in a position such that the mechanical (load-bearing) axis is vertical. In the case of a normal knee this axis is a line extending from the center of rotation of the femoral head to the center of the knee joint passing distally to bisect the ankle joint [12, 13]. A local coordinate system - the orthogonal basis of the femur will be formed by the mechanical, sagittal and transverse axes with its origin at the knee-joint center. Two major problems are the computation of the centers of the femoral head and of the knee joint. We start with the

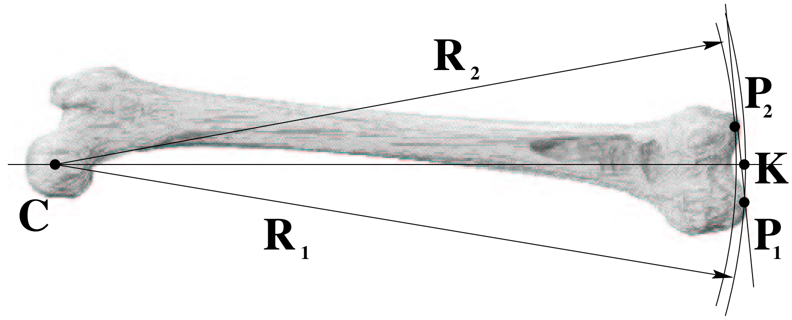


Figure 3: Searching for the knee joint center

former.

2.1. Center of Femoral Head

Although the femoral head is not absolutely spherical and the exact shape is still a subject of controversy [6, 9, 10], we assume with satisfactory accuracy that part of the head is a sphere, whose center is sought.

Obviously, this point cannot be found precisely and a numerical method is to be devised to compute the desired center as closely as possible. The proposed algorithm for computing the head's center starts with the separation of the set of points, $\{P_i\}$, which are actually part of the spherical head, $\mathcal{S}$. Toward this end, a proper cutting plane p is selected (see Figure 2) and the bone is separated along this plane. Thereafter, only the set of points, $\psi = \{P_i | P_i \in \mathcal{S}, \forall i\}$, is considered.

One could apply the following approach for seeking the desired center. Let $C = (x_C, y_C, z_C)$ be the center of the head. Then, for any point $P_i = (x_{P_i}, y_{P_i}, z_{P_i}), P_i \in \mathcal{S}$, the following constraint has to be satisfied:

$$(x_C - x_{P_i})^2 + (y_C - y_{P_i})^2 + (z_C - z_{P_i})^2 \quad = \quad \mathcal{R}^2, \;\; \forall i, \tag{1}$$

where $\mathcal{R}$ is the radius of the sphere. Equation(1) develops into a system of quadratic equations in four unknowns: x_C, y_C, z_C and $\mathcal{R}$.

We convert the problem into a set of linear constraints. Let $Q = (x_Q, y_Q, z_Q), Q \in \mathcal{S}$ be the point with the largest distance from plane p (see Figure 2). Denote by $M_i = (x_{M_i}, y_{M_i}, z_{M_i})$ the middle point of the line segment $\overline{P_iQ}, P_i \in \mathcal{S}$. Then, the segments $\overline{P_iQ}$ and $\overline{CM_i}$ have to be orthogonal. The inner product of the two vectors, $\overrightarrow{P_iQ}$ and $\overrightarrow{CM_i}$, must equal zero:

$$(x_Q - x_{P_i})(x_{M_i} - x_C) + (y_Q - y_{P_i})(y_{M_i} - y_C) + (z_Q - z_{P_i})(z_{M_i} - z_C) = 0. \tag{2}$$

Denote by $\tilde{\psi} = \psi \setminus \{Q\}$ the set of all points in $\mathcal{S}$ except Q and let,

$$b_i = x_{M_i}(x_Q - x_{P_i}) + y_{M_i}(y_Q - y_{P_i}) + z_{M_i}(z_Q - z_{P_i}), \;\; \forall i.$$

The following system of linear equations in (x_C, y_C, z_C),

$$\left\{ \begin{array}{rcl} x_C(x_Q - x_{P_1}) + y_C(y_Q - y_{P_1}) + z_C(z_Q - z_{P_1}) & = & b_1 \\ x_C(x_Q - x_{P_2}) + y_C(y_Q - y_{P_2}) + z_C(z_Q - z_{P_2}) & = & b_2 \\ & \vdots & \\ x_C(x_Q - x_{P_n}) + y_C(y_Q - y_{P_n}) + z_C(z_Q - z_{P_n}) & = & b_n \end{array} \right. \tag{3}$$

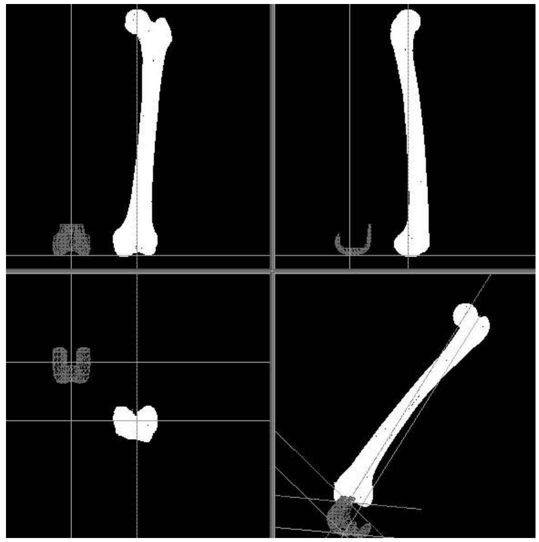


Figure 4: Implant and femur with separate coordinate systems.

is obtained.

Evidently, for any real data set, the number of points, n, exceeds the number of unknowns. Thus, one needs to find the optimal center (x_C, y_C, z_C), under the constraint problem of (3). On the other hand, the problem can be considered as a least squares problem with respect to the orthogonality of the $\overrightarrow{P_iQ}$ and $\overrightarrow{CM_i}$. Here, the singular value decomposition (SVD) method [11] is employed to solve this least squares problem of (3).

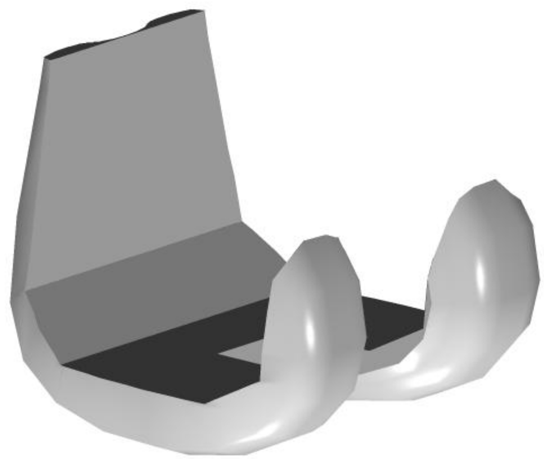
Finally, while the farest point Q was selected due to stability considerations, one should recall that depending on the desired accuracy, several sufficiently far Q points from the plane can be selected, reducing the error that can be introduced due to the dependency on a single Q point.

2.2. Center of Knee Joint

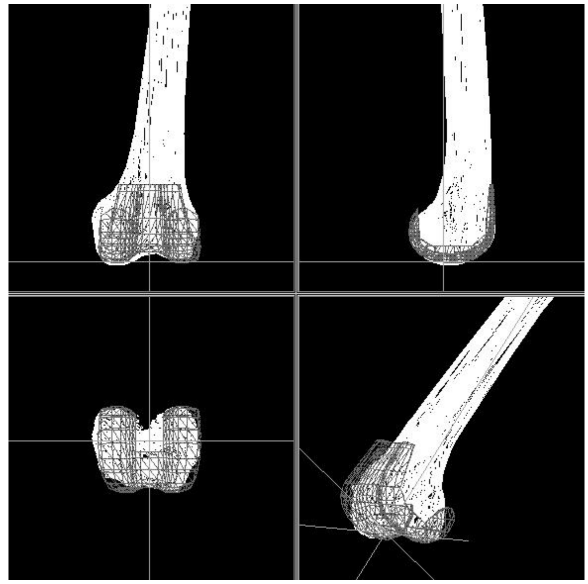
The next step is the computation of the center of the knee joint. We propose to determine the location of the two extreme points of the femoral condyles, P_1 and P_2, (see Figure 3). In an upright position these points are the centers of load transfer regions from the femur to the tibia. Then, the knee joint's center, K, can be computed as a center point between P_1 and P_2. An iterative method was used for searching these two extreme points.

Construct a sequence of concentric spheres centered at C with increasing radii. These spheres can have none, one or two intersection contours with the femur. Although spherical, assume for simplicity, that the intersection contour(s) are piecewise linear planar closed curves. Then, the area inside the contour can be easily computed by the trapezoid method. That is, for each linear segment of the given contour compute the signed area of the trapezoid between the segment and an auxiliary axis. The sum of all areas is the contour's area up to the sign.

If the sphere is tangent to one condyle, the point of tangency could be considered the desired extreme point. Unfortunately, it is very difficult to pinpoint this tangency precisely. In practice, we decide that tangency occurs when the area inscribed by the intersecting contour is less than some prescribed tolerance, and compute the tangent point as the center of all points on the contour. To converge to the tangent point, a bisection method on the radius of the sphere is used.



(a) femoral implant component



(b) superimposition of bone/implant

Figure 5: Implant positioning at total knee replacement.

Once the center of the femoral head, C, and the center of the knee joint, K, are known, the line $\overline{CK}$ is defined as the mechanical bone axis. In most cases the line $\overline{P_1P_2}$ is not orthogonal to the mechanical axis $\overline{CK}$. Denote the plane through the points P_1, P_2, and C - the frontal plane of the femur. Then rotate the line $\overline{P_1P_2}$ in this frontal plane about point K until it becomes perpendicular to the mechanical axis. The obtained line is defined to be the transverse axis. Finally, the sagittal axis, perpendicular to the mechanical and transverse axes, is computed as the cross product of two unit vectors collinear with the mechanical and transversal axes.

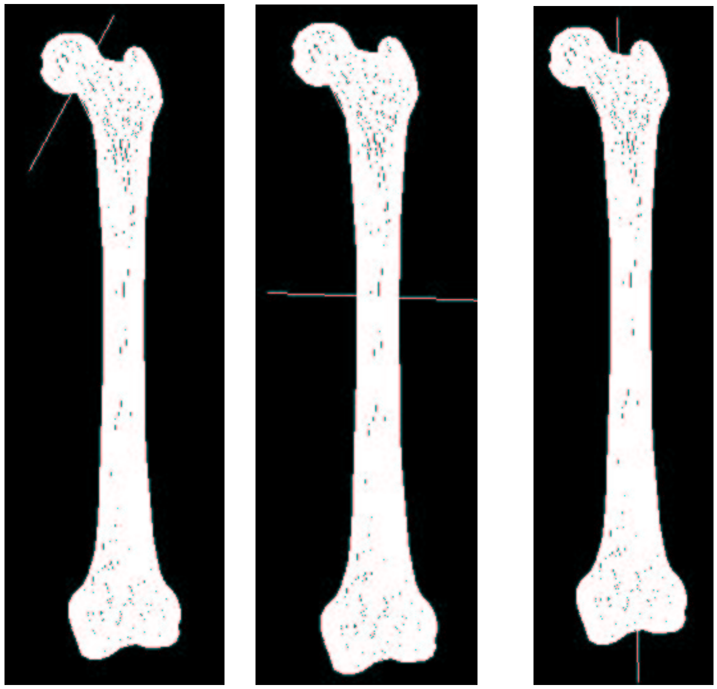
Once the computation of the local axes of the femur is complete, superpositioning of an implant onto the femur can be automatically performed by bringing the local axes of the prosthesis (see Figure 4) to be coincident with the axes of the orthogonal system.

In Figure 5 (a), we see a simplified computer model of an implant that was reverse-engineered from drawings of an implant provided by the implant manufacturer. In Figure 5 (b), three projections and an isometric view of the implant superimposed on the bone are shown. Corrections, if needed, can be performed by the surgeon, interactively.

3. Efficient Plane - Object Intersections

To perform the bisection of the femur along plane p, as shown in Figure 2, one needs to find the intersection of the bone with a given cutting plane as well as to divide the model into two separated parts. Large amount of data in the model of the femur can hinder interactive processing of such an intersection. To accelerate the process, an octree data structure (also called Octree Encoding) can be used. The octree data structure allows one to execute the search in a sublinear time, where n is the number of polygons in the femur.

The fundamental power of the octree representation stems from its 'divide-and-conquer' binary subdivision. An octree is derived by successively subdividing the Euclidean space in all three dimensions by parallel planes, forming cubes. Since the original 3D medical data set has a voxel representation, it is very suitable for use within an octree representation for data



(a) 7310 polygons processed, 325 intersected.

(b) 5444 polygons processed, 265 intersected.

(c) 20616 polygons processed, 1168 intersected.

Figure 6: Speed-up results in plane-model intersection for the model of a femur consisting of 31263 polygons. Speed-up is measured as number of polygons processed and number of polygons split.

storage and manipulation.

In addition to its spatial characteristics and subtrees, each node of the octree contains two lists: one of polygons entirely contained in the corresponding cube, the 'Full' list, and a second list of polygons only partially contained in this cube, the 'Partial' list. Creation of such a data structure requires spatial sorting. To build this octree encoding, one starts from the top or the entire universe and relegate to the 'Full' list of the root all polygons of the model that are completely contained in the root's universe. The 'Partial' list is empty. This top cube completely encloses the entire object. Then, a recursive algorithm of spatial polygon sorting is applied. The location of each polygon from the 'Full' list is verified against the eight sub-cubes of the octree. If the polygon is entirely contained in one sub-cube, this polygon is relegated to the 'Full' list of that sub-node. Otherwise, if the polygon shares some sub-cubes, it is included in the several corresponding 'Partial' lists. Once all polygons in the current node are traversed, spatial subdivision continues recursively for all subtrees until a minimum size cube is reached.

With the octree's spatial subdivision of the model as a preprocessing stage, one can perform efficient intersection operations with arbitrary cutting planes. Here, we present a fast algorithm for finding the contour(s) of intersection of the model of the bone with some arbitrary plane, as well as for dividing the model into two separated parts. Each of the two parts will be represented by a list of polygons, that are located all to the left (or right) of

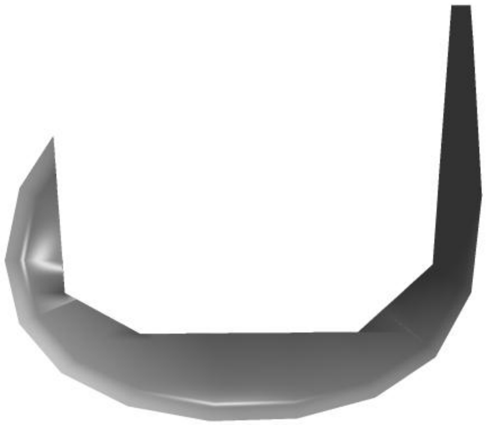


Figure 7: The femur after being shaped for the implant.

the given plane. In general, such subdivision implies processing of each and every polygon of the model for its location with respect to the plane, or alternatively, whether it crosses the plane. Obviously, such traversal is linear in the input size and required large amount of time for models with large number of polygons. Nevertheless, the octree data structure allows one to reduce this time to a sublinear order.

The proposed algorithm processes nodes of the octree recursively, starting from the top of the tree. For each of the eight subtrees, the location of the corresponding sub-cube with respect to the cutting plane, p, is verified. If p does not intersect the sub-cube, the status of all the polygons in the 'Full' list of this sub-cube with all its descendents is completely determined and no further processing is necessary. The search continues recursively only for those sub-cubes which do intersect with p. The segments of the contour of intersection, if any, are computed for the polygons contained in the sub-cubes that intersected p.

In general, the cutting plane can have an arbitrary orientation with respect to the bone. In the worst possible case, when the cutting plane intersects with all polygons of the model, the process is reduced to a linear time. In our example (Figure 6), the extreme case is (c). However, for most practical cases the zones of intersection we are interested in are far less extreme (cases (a) and (b) in Figure 6). These areas require processing of only a small number of the polygons. For these cases, the presented algorithm requires a sublinear computation time, processing less than one-fourth of the data. Hence, cutting operations in interactive surgery planning is significantly more practical. Figure 7 shows five sections along the five cutting planes of the implant. Another example for the need for a cutting operation is oblique osteotomy discussed in the next section.

4. 3D Oblique Osteotomy

Once the local coordinate system is fixed, the anatomical axis of the femur can easily be computed. Using the above-stated intersection algorithm one can interactively perform a preoperative 3D oblique osteotomy planning. First, a single cut of the femur in its proximal

(a) rotation

(b) rotation and translation

Figure 8: 3D femoral oblique osteotomy.

part is computed. Then, rotation of the proximal part of the femur relative to the distal part, through a certain angle about an axis perpendicular to the cutting plane, is performed, followed by possible relative translation in the cutting plane (see Figure 8).

The theory and the practice of 3D oblique osteotomy, including the specific geometric orientation problems of getting the position of the intersection plane, the axis and angle of rotation and the translation were discussed in [1, 2, 3, 4, 5, 6]. Based on the solutions given there, special measuring (under fluoroscopy) and operational instrumentation and techniques were developed. 14 operations were performed in Haifa and Johannesburg with encouraging results.

The shared contact area of the bone parts after the rotation and translation, T, should be at certain ratio, ρ, to the area bounded by the contour of the bone cut, A. Denote the area of the section after rotation and translation by $\tilde{A}$, then $T = A \cap \tilde{A}$ and $\rho = \frac{T}{A}$ is recommended to be above 60%.

To satisfy this percentage requirement, one needs to compute the ratio of ρ interactively during the rotation and translation to ensure that it remains within the given percentage. Unfortunately, the obtained contour of intersection of the femur with the cutting plane, p, contains a significant amount of "noise". To avoid possible computational error and enhance the robustness of the analysis, we derive the convex hull of all points, in the contour of intersection, using Graham's algorithm [7]. Then, the areas of the obtained convex hulls are computed in a similar way to the one described in the section 2.

Acknowledgment

We are thankful to Michael Lindenbaum (Technion) for his contribution to the solution of the problem of finding the center of the femoral head. We are also thankful to Elscint for allowing us to use their equipment to perform the CT scan and to BIOMET Inc. for providing the drawings and specifications of the implant.

References

[1] Y. Charit, M. Roffman: *Experimental and Clinical Experience with Oblique Osteotomy for Perthes Disease*. WITS, ISBN 0 85494 7000, Johannesburg 1981.

[2] Y. Charit: *The Geometry of Oblique Osteotomy*. Eng. Design Graphics J. **46**, No. 3, 22-25 (1982).

[3] Y. Charit: *A Computerized Geometrical Analysis and Solution for a Surgical Problem*. Proc. Internat. Conference on Engineering and Computer Graphics, Beijing 1984, 231-236.

[4] Y. Charit, G.T. Du Toit: *The Problem and Theoretical Solution of a Three-Dimensional Realignment of Deformed Bones*. WITS, ISBN 0 85494 939 9, Johannesburg 1986.

[5] Y. Charit, M. Eck, J. Hoschek et al.: *Geometric Problems in Performing Oblique Osteotomies*. Technische Hochschule Darmstadt, Preprint #1638 (1994).

[6] M. Eck, J. Hoschek, U. Weber: *Three-dimensional Determination of an Oblique Osteotomy*. J. Biomechanics **23**, 1061-1067 (1990).

[7] R.L. Graham: *An Efficient Algorithm for Determining the Convex Hull of a Planar Set*. Information Processing Letters, North-Holland Publ. Comp. 1972, 132-133.

[8] W.E. Lorensen, H.E. Cline: *Marching Cubes: A High Resolution 3D Surface Construction Algorithm*. Computer Graphics (SIGGRAPH'87 Proc.) **21**, No. 4, 163-169 (1987).

[9] A. Menschik: *Biometrie*. Springer Verlag, Berlin 1987, 231-232.

[10] J. Mizrahi: *The Human Femoral Head as a Tilted Solid of Revolution*. South African Journal of Science **73**, p. 280 (1977).

[11] W.H. Press, B.P. Flannery, S.A. Teuolsky, W.T. Vettering: *Numerical Recipes in C*. Cambridge University Press, pp. 60-72.

[12] Th.P. Sculco: *Primary Total Knee Replacement*. Chapter 30 in W. Petty, Total Joint Replacement, Sounders Company 1991, p. 513.

[13] A. Steindler: *Mechanics of Normal and Pathological Locomotion in Man*. Bailliere, Tindal & Cox ed., London 1935.

Received November 26, 1996; final form May 29, 1997